

EUROPEAN PATENT OFFICE

Patent Abstracts of Japan

PUBLICATION NUMBER : 04365709
PUBLICATION DATE : 17-12-92

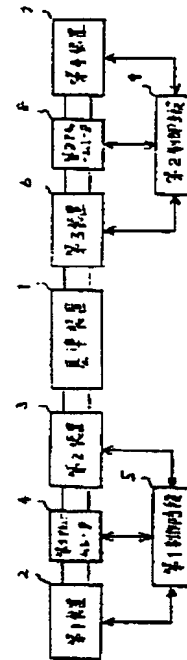
APPLICATION DATE : 27-11-90
APPLICATION NUMBER : 02323758

APPLICANT : MITSUBISHI MATERIALS CORP;

INVENTOR : TAGAO MASAMI;

INT.CL. : B65G 43/08 B21D 51/26 G05B 13/02

TITLE : PROCESSING SPEED CONTROL
SYSTEM FOR CAN MANUFACTURING
LINE



ABSTRACT : PURPOSE: To quickly and precisely set the processing speed by making the target processing speed of the first device coincide with the actual processing speed of the second device if the number of staying half-finished goods of the second device upstream from a reference device is within a proper range, and conducting the similar processing in the downstream of the reference device.

CONSTITUTION: The first device 2 and the second device 3 are provided upstream from a reference device 1, and the number of half-finished goods in the first accumulator 4 for once storing half-finished goods just before the second device 3 is monitored by the second control means 5. If the number of staying half-finished goods is within a proper range, the target processing speed of the first device 2 is made agree with the actual processing speed of the second device 3. The third device 6 and the fourth device 7 are disposed downstream from the reference device 1, and the number of staying half-finished goods in the second accumulator 8 for once storing the half-finished goods just before the fourth device 7 is monitored by the second control means 9. If the staying number is within a proper range, the target processing speed of the fourth device 7 is made to coincide with the actual processing speed of the third device 6.

COPYRIGHT: (C)1992,JPO&Japio

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平4-365709

⑬ Int. Cl.⁵

B 65 G 43/08
B 21 D 51/26
G 05 B 13/02

識別記号

B 9245-3F
Z 7011-4E
Z 9131-3H

庁内整理番号

⑭ 公開 平成4年(1992)12月17日

審査請求 未請求 請求項の数 1 (全23頁)

⑮ 発明の名称 製缶ラインの処理速度制御システム

⑯ 特 願 平2-323758

⑰ 出 願 平2(1990)11月27日

⑱ 発 明 者 岩 瀬 実 茨城県結城市新堤仲通り1番地の1 三菱金属株式会社結城工場内
⑱ 発 明 者 田 原 正 美 茨城県結城市新堤仲通り1番地の1 三菱金属株式会社結城工場内
⑲ 出 願 人 三菱マテリアル株式会 東京都千代田区大手町1丁目5番1号
社
⑳ 代 理 人 弁理士 桑井 清一 外1名

明 細 書

1. 発明の名称

製缶ラインの処理速度制御システム

2. 特許請求の範囲

複数の装置で構成された製缶ラインの処理速度を制御する製缶ラインの処理速度制御システムにおいて、

上記複数の装置のうちのひとつの装置が基準装置として選択されており、

上記基準装置より上流側に配置され少なくとも2つの工程がそれぞれ割り当てられた第1および第2装置と、

第2装置の直前で半製品を一旦貯めておく第1アキュムレータと、

第1アキュムレータ中の半製品の滞留数を監視して該滞留数が適正範囲なら、第1装置の目標処理速度を第2装置の実処理速度に一致させる第1制御手段と、

上記基準装置より下流側に配置され少なくとも

2つの工程がそれぞれ割り当てられた第3および第4装置と、

第4装置の直前で半製品を一旦貯めておく第2アキュムレータと、

第2アキュムレータ中の半製品の滞留数を監視して、該滞留数が適正範囲なら第4装置の目標処理速度を第3装置の実処理速度に一致させる第2制御手段とを有することを特徴とする製缶ラインの処理速度制御システム。

3. 発明の詳細な説明

<産業上の利用分野>

本発明は製缶工場の製缶ラインに係り、特に、互いに処理速度の異なる一連の処理装置を制御する処理速度制御システムに関する。

<従来の技術>

従来から製缶工場における製缶ラインでは、金属の薄板の打ち抜きから完成した缶胴を出荷するまでを一連の構成装置で連続して処理していた。

詳述すると、製缶ラインは金属の薄板からカップを打ち抜くカッピングプレス装置と、上記カップを深絞り加工して所定長の缶胴を成形する深絞り装置と、上記缶胴を洗浄する洗浄装置と、洗浄された缶胴の外面に所定の模様を印刷する印刷装置と、印刷された模様を固定させるオープン装置と、缶胴の内面処理をする内面処理装置と、缶胴の開口部を加工する開口部加工装置と、缶胴の不良の有無を検査する検査装置と、検査後の缶胴を出荷する出荷装置とを含んでおり、これらの装置は搬送ラインで接続され、缶胴は前段の処理が終了すると搬送ラインで次段の処理をおこなう装置へ搬送される。

各構成装置にはそれぞれ制御部（ビルトインコンピュータ）が設けられており、オペレータはこれらのビルトインコンピュータを個別に操作して単時間当りの目標処理数を与えていた。したがって、これらの構成装置はビルトインコンピュータから与えられた目標にしたがって自らの処理速度を制御していた。

ータは個々の構成装置の処理速度を最適に調整できるとは限らないので、素早く全装置の再設定を行うことができない。その結果、トラブルに対応する的確な処理が困難であるという問題点もあった。

そこで、本発明の目的は、搬送ラインを構成する各装置の処理速度の設定作業を生産コストを上げることなく、しかも製缶ラインの稼働率を低下させることなく行うことができるとともに、素早く、しかも的確に行うことができる製缶ラインの処理速度制御システムを提供することである。

＜課題を解決するための手段＞

本発明は、第1図に示すように複数の装置で構成された製缶ラインの処理速度を制御する製缶ラインの処理速度制御システムにおいて、上記複数の装置のうちのひとつの装置が基準装置1として選択されており、上記基準装置1より上流側に配置され少なくとも2つの工程がそれぞれ割り当てられた第1装置2および第2装置3と、第2装置

また、上述のように、製缶ラインは複数の装置で構成されているが、製缶ライン全体の処理速度と各構成装置間の相対的処理速度は、オペレータが運転状況を見ながら経験則にてらして調整していた。

＜発明が解決しようとする課題＞

しかしながら、オペレータがこれらの装置の単時間当りの目標処理数を設定する場合、製缶ライン全体の処理速度と各構成装置間の相対的処理速度を監視しながらビルトインコンピュータを操作していたので、各装置の処理状況を監視するオペレータを多数必要とし、生産コストが上昇するという問題点があった。

また、オペレータが手動でビルトインコンピュータを操作していたので、設定変更作業に多くの時間を要し、製缶ラインの稼働率が低いという問題点があった。

また、例えば缶のクラッシュ等のトラブルによって一部の装置の処理速度が低下しても、オペレ

3の直前で半製品を一旦貯めておく第1アキュムレータ4と、第1アキュムレータ4中の半製品の滞留数を監視して該滞留数が適正範囲なら、第1装置2の目標処理速度を第2装置3の実処理速度に一致させる第1制御手段5と、上記基準装置1より下流側に配置され少なくとも2つの工程がそれぞれ割り当てられた第3装置6および第4装置7と、第4装置7の直前で半製品を一旦貯めておく第2アキュムレータ8と、第2アキュムレータ8中の半製品の滞留数を監視して、該滞留数が適正範囲なら第4装置7の目標処理速度を第3装置6の実処理速度に一致させる第2制御手段9とを有する製缶ラインの処理速度制御システムである。

＜作用および効果＞

上記構成に係る処理速度制御システムを使用し、複数の装置で構成された製缶ラインの処理速度を制御するには、まず、オペレータは複数の装置の中からひとつの装置を基準装置として選択し、

その基準装置に目標処理速度のデータを供給する。

次に、基準装置より上流側では第1制御手段が基準装置の目標処理速度のデータに基づいて各装置に目標処理速度のデータを供給する。一方、基準装置より下流側では第2制御手段が基準装置の目標処理速度のデータに基づいて各装置に目標処理速度のデータを供給し、各装置を稼働させる。

そして、第1制御手段は第1アキュムレータ中の半製品の滞留数を監視して該滞留数が適正範囲なら、第1装置の目標処理速度を第2装置の実処理速度に一致させる。

また、第2制御手段は第2アキュムレータ中の半製品の滞留数を監視して、該滞留数が適正範囲なら第4装置の目標処理速度を第3装置の実処理速度に一致させる。

このように、第1装置、および第2装置に各装置の処理状況を監視させているため、オペレータの数を減らすことができ、その結果、生産コストを下げる事ができる。また、第1装置、および第2装置に各装置の処理速度の設定変更作業を行

なわさせているため、処理速度の設定変更作業の時間を短縮させることができ、その結果、製缶ラインの稼働率を高めることができる。さらに、第1装置、および第2装置に各装置の処理状況を監視、ならびに処理速度の設定変更作業を行なわさせているため、上述したように缶のクラッシュ等のトラブルによって一部の装置の処理速度が低下しても素早く全装置の再設定を行うことができるので、その結果、トラブルに対応する的確な処理が容易である。

〈実施例〉

以下、本発明に係る実施例を図面を参照して説明する。

第2A図～第2C図は本願発明の一実施例を説明する概略側面図であり、第2A図の右側に第2B図が置かれ、第2B図の更に右側に第2C図が続くと、本実施例の製缶ラインの全体構成が明らかになる。

第2A図において、11はアルミニウム製の薄

板(厚さ約0.3ミリメートル～0.35ミリメートル)のロール12、13を巻戻し、打ち抜き装置としてのカップングプレス14に連続して供給するアンワインダであり、カップングプレス14はアルミニウムの薄板を打ち抜き、直径7.5センチメートル～8.9センチメートル、深さ約3センチメートルのカップを成形する。

カップングプレス14で連続的に成形されるカップは簡単な水平搬送装置15でドロアンドアイオニングプレス16に供給され、深絞り加工により缶胴に成形される。缶胴はトリマ17で深さを調整され、その後、洗浄装置としてのウオッシャ18に簡単な搬送ラインで運ばれる。なお、本実施例では、ドロアンドアイオニングプレス16とトリマ17とで深絞り装置を構成している。

水平搬送装置15は、第3図に概略図示されているように、X方向に缶胴19を空気の噴流で移動させ、缶胴19を幅広のアキュムレータ20に一旦保持し、その後、排出部21から1個毎に搬送する。上記アキュムレータ20には、第2

A図に図示されているように、空気の噴流量を調節する駆動ユニット85と上記アキュムレータ20に貯留される単時間当りの缶胴数を計測するセンサ86とが設けられている。したがって、下記に示すスレーブコンピュータは上記アキュムレータ20の稼働時間とセンサ86から供給される単時間当りの缶胴数とから上記アキュムレータ20を通過した缶胴19の数を把握することができる。

ウオッシャ18では、缶胴19は酸洗、および耐蝕性向上のための表面処理を施され、その後、加熱乾燥される。かかる一連の処理を受けた缶胴19は、ウオッシャ18から垂直搬送装置22、および水平搬送装置23を経てベースコート24に供給される。この水平搬送装置23には水平搬送装置15と同様にアキュムレータ45(図示省略)が設けられているとともに、該アキュムレータ45には駆動ユニット87とセンサ88とが設けられている。

ベースコート24では、缶胴19の外表面が一樣

に白色塗装され、その後、これら白色塗装された缶胴19はベースコータオープン25に簡単な搬送ライン26で運ばれる。ベースコータオープン25は缶胴19の外面に塗装された白色塗料の焼付け、乾燥を行い、その後、缶胴19を水平搬送装置27でプリンタ28に供給する。この水平搬送装置27には水平搬送装置15と同様にアキュムレータ(図示省略)が設けられているとともに、該アキュムレータには駆動ユニット89とセンサ90とが設けられている。

プリンタ28は白色塗装された缶胴19の外面に文字、および、または図形等を印刷し、印刷の済んだ缶胴19は簡単な搬送ライン29でプリンタオープン30に供給される。このプリンタ28は基準装置として指定されているので、各装置の目標処理速度のデータはこのプリンタ28の目標処理速度のデータに基づいて計算され、供給される。

プリンタオープン30は印刷された塗料を焼き付け、乾燥し、文字、図形等となる模様を固定す

面処理装置を構成している。

インサイドスプレイオープン36から排出された缶胴19は、簡単な搬送ライン37で開口部加工装置としてのネッカーフランジ38に運ばれ、そこで、その開口部に多段の絞り加工が施される。その後、缶胴19は簡単な搬送ライン39と水平搬送装置40とを介して検査装置としてのライトテスト41に移送され、ライトテスト41はピンホール等を光学的に検出して不良品を排除する。この水平搬送装置40には水平搬送装置15と同様にアキュムレータ(図示省略)が設けられているとともに、該アキュムレータには駆動ユニット93とセンサ94とが設けられている。

かようにして検査済みの缶胴19はライトテスト41から出荷装置としてのパレタイザ42に送られ、プラスチックフィルムで密封された状態で荷造りされ、出荷される。

上述の製缶ラインを構成する各装置11、14、16、18、24、25、28、30、34、36、38、41、42は内蔵コンピュータ(ビル

ドインコンピュータ)51、52、53、54、55、56、57、58、59、60、61、62、63をそれぞれ備えている。

また、これらのビルトインコンピュータ51～63は操作パネル64、65、66、67、68、69、70、71、72、73、74、75、76を有しており、オペレータがこの操作パネル64～76を操作することにより、各装置毎の処理速度を含む稼働条件を指示することもできる。

本実施例に係る製缶ラインの構成装置は水平搬送装置23により2つの構成装置群に分けられており、それぞれの構成装置群は情報処理装置としてのスレーブコンピュータ77、78によっても運転状態が自動的に管理される。また、スレーブコンピュータ79は各水平搬送装置15、23、27、33、40のアキュムレータで保持している缶胴19の数を把握し管理している。

すなわち、スレーブコンピュータ77は内蔵コンピュータ51～54と外部バス80を介して接続されており、装置11、14、16、18から

トインコンピュータ)51、52、53、54、55、56、57、58、59、60、61、62、63をそれぞれ備えている。

また、スレーブコンピュータ79は各水平搬送装置15、23、27、33、40のアキュムレータで保持している缶胴19の数を把握し管理している。

すなわち、スレーブコンピュータ77は内蔵コンピュータ51～54と外部バス80を介して接続されており、装置11、14、16、18から

【表1】

装置	レポートデータ	データメモリ	記号
キャビン コンソール 14 (C P)	運転開始後の 総処理数	→	TL _{co}
	毎分の処理 数	→	CPM _{co}
インフォ マティクス コンソール 16 (D I)	運転開始後の 総処理数	→	TL _{di}
	毎分の処理 数	→	CPM _{di}
ワッチ 18 (W S)	今の処理数 1分前の処 理数 ・ ・ ・ 10分前の 処理数	→	CPM _{ws} CPM _{ws1} ・ ・ ・ CPM _{ws10}

なお、ホストコンピュータ83はタイマーで一定時間が経過するとスレーブコンピュータ77にレポートの送付を要求し、スレーブコンピュータ77からのレポートデータを受け取り、後述するように各構成装置の運転条件を示すオペレーションデータを発生させる。

また、スレーブコンピュータ77はホストコン

の処理状況を示すデータを受け入れ、これらのデータに基づき処理状況を監視し制御する。

また、スレーブコンピュータ77は上記装置になんらかの不具合が生じた場合、オペレータにその不具合を知らせるために上記装置にアラームを発生させる。例えば、オペレータにアンワインダ11のロール12、13の交換時を知らせるためにアンワインダ11の操作パネル64にアラームを発生させる。

更に、スレーブコンピュータ91は以下の表1で示すレポートデータを作成し、このレポートデータを下記に示すホストコンピュータ83のデータメモリ103へ供給する。なお、CPMは毎分処理される毎瞬数を表している。

(以下、余白)

ビュータから送られてくるオペレーションデータに基づきビルトインコンピュータ用の命令コードを作成し、必要なビルトインコンピュータ51～54に各装置の運転条件を表す命令コードを供給する。

同様に、スレーブコンピュータ78は内蔵コンピュータ55～63に外部バス81を介して接続されており、装置24、25、28、30、34、36、38、41からの処理状況を示すデータを受け入れ、これらのデータに基づき処理状況を監視する。また、スレーブコンピュータ78はスレーブコンピュータ78と同様に上記装置になんらかの不具合が生じた場合、オペレータにその不具合を知らせるために上記装置の操作パネルにアラームを発生させる。更に、スレーブコンピュータ78は以下の表2で示すレポートデータを作成し、このレポートデータを下記に示すホストコンピュータへ供給する。

(以下、余白)

【表2】

装置	レポートデータ	データメモリ	記号
バック アップ 24 (B S)	運転開始後の 総処理数	→	TL _{bs}
	毎分の処理 数	→	CPM _{bs}
プリンタ 28 (P R)	運転開始後の 総処理数	→	TL _{pr}
	毎分の処理 数	→	CPM _{pr}
インサ ート 34 (I N)	運転開始後の 総処理数	→	TL _{in}
	毎分の処理 数	→	CPM _{in}
インサ ート 36 (I O)	今の処理数 1分前の処 理数 ・ ・ ・ 10分前の 処理数	→	CPM _{io} CPM _{io1} ・ ・ ・ CPM _{io10}
バック アップ 38 (N F)	運転開始後の 総処理数	→	TL _{nf}
	毎分の処理 数	→	CPM _{nf}

ライトテスト 41	運転開始後 の総処理数	→	TLit
(LT)	毎分の処理 数	→	CPMit

なお、ホストコンピュータ83はタイマーで一定時間が経過するとスレーブコンピュータ78にもレポートの送付を要求し、スレーブコンピュータ78からのレポートデータを受け取る。ホストコンピュータ83は各構成装置の運転条件を表すオペレーションデータを後に詳述するように作成する。

また、スレーブコンピュータ78はホストコンピュータから送られてくるオペレーションデータに基づきビルトインコンピュータ用の命令コードを作成し、必要なビルトインコンピュータ55～63に運転条件を示す命令コードを供給する。

一方、スレーブコンピュータ79は各駆動ユニット85、87、89、91、93と各センサ86、88、90、92、94とに外部バス82を

介して接続されており、各水平搬送装置15、23、27、33、40のアキュムレータ内の缶胴19の数を示すデータを示すデータを受け入れ、以下の表3で示すレポートデータを作成し、このレポートデータを下記に示すホストコンピュータへ供給する。

(以下、余白)

「表3」

装置	レポート	データ	記号
水平搬送装置15の7キュームレ-9	滞留数	→	Ac1
水平搬送装置23の7キュームレ-9	滞留数	→	Ac2
水平搬送装置27の7キュームレ-9	滞留数	→	Ac3
水平搬送装置33の7キュームレ-9	滞留数	→	Ac4
水平搬送装置40の7キュームレ-9	滞留数	→	Ac5

これらのスレーブコンピュータ77～79は、第2A図～第2C図に示すように、更に上位のホストコンピュータ83と外部バス84を介して接続されており、ホストコンピュータ83の支配下にある。

第4図は本発明の一実施例に係る製缶ラインの制御システムのホストコンピュータのハードウェアの構成を示すブロック図である。

この図のホストコンピュータは、中央処理装置(以下、CPUという)101とプログラムメモリ102とデータメモリ103とビデオメモリ104とグラフィックコントローラ105とDMAコントローラ106とインターフェース107等とから構成されている。

CPU101はプログラムメモリ102内の命令コードを随次フェッチし、与えられたJOBを遂行する。システム起動時には、CPU101はプリンタ108の目標値が与えられると、カップングプレス14、ドローアンドファイオニングプレス16、ウォッシャ18、ベースコート24、インサイドスプレイ34、ネッカーフランジ38、ライトテスト41の標準値を決める。また、システムの稼働中には、CPU101は各スレーブコンピュータ77～79からのレポートデータに基づき各装置の運転条件(例えば、処理速度)を示

すオペレーションデータを作成し、データメモリ103に保持する。

ビデオメモリ104はディスプレイ109に表示するメッセージを表すデータ(メッセージデータ)を書換え可能に記憶する。なお、書換えはCPU101がデータメモリ103からメッセージデータをビデオメモリ104に転送する。

グラフィックコントローラ105はCPU101からのリクエストによりメッセージデータをビデオメモリ104から順次読み出しディスプレイ109に表示する。

DMAコントローラ106はCPU101からのリクエストにより3台のスレーブコンピュータ77~79から定期的に送られてくる各装置の処理状況を示すレポートデータをインターフェース107からデータメモリ103に転送する。また、DMAコントローラ106はCPU101からのリクエストによりオペレーションデータをインターフェース107を介して指定されたスレーブコンピュータ77~79に転送する。

行し、ホストコンピュータ83とスレーブコンピュータ77~79はそれぞれのメモリ等を初期化する。この初期化において、IDXフラグとEMフラグとも初期化する(ステップS1)。IDXフラグはスレーブコンピュータ77~79のいずれかを指定するフラグであり、「0」の時はスレーブコンピュータ77を、「1」の時はスレーブコンピュータ78を、「2」の時はスレーブコンピュータ79を指定する。また、EMフラグはプリンタ28が異常状態「1」であるか否か「0」を示すためのフラグである。詳しくは、ホストコンピュータ83は基準装置であるプリンタ28の上流側に位置する水平搬送装置27のアキュムレータ、または下流側に位置する水平搬送装置33のアキュムレータに蓄えられている缶の数が適正範囲外であれば、システムに異常状態が発生したと判断して、EMフラグに「1」をセットする。一方、システムが正常状態であると判断した場合は、EMフラグに0をセットする。これらのフラグ設定は後に詳述する。

インターフェース107は3台のスレーブコンピュータ77~79からのレポートデータを一時記憶した後、DMAコントローラ106のアドレス発生に同期してこのレポートデータをデータメモリ103に転送する。また、インターフェース107はCPU101から転送される搬送ラインの処理状況を示すデータをプリンタ108に転送し、レポートを作成させる。更に、インターフェース107はデータメモリ103から送られてきたオペレーションデータを一時記憶し、その後、指定されたスレーブコンピュータ77~79へ送信する。

次に、上記ホストコンピュータ83の動作手順を第5A図~第5B図および第6図~第15図のフローチャートを用いて説明する。

まず、第5A図~第5B図はホストコンピュータ83のメインフローチャートを示している。図にあって、オペレータがシステムを起動させると、ホストコンピュータ83がシステムの初期化を実

次に、ホストコンピュータ83はオペレータが基準装置であるプリンタ28に目標処理速度のデータを供給したか否かを判断する(ステップS2)。その判断結果がNOならば、オペレータがプリンタ28に目標処理速度のデータを供給していないので、オペレータがプリンタ28に目標処理速度のデータを供給するまでステップS2を繰り返し実行する。

一方、ステップS2の判断結果がYESならば、ホストコンピュータ83のCPU101は、ステップ2で与えられた上記プリンタ28の目標処理速度のデータを参照して以下に示す各処理装置の標準処理値を作成する(ステップS3)。すなわち、カップングプレス14(CP)、ドローアンドアイオニングプレス16(DI)、ウオッシュ18(Ws)、ベースコート24(Bs)、インサイドスプレイ34(IN)、ネッカーフランジ38(NF)、ライトテスト41(LT)の標準処理値がプリンタ28の目標処理値に連動して決定される。

そして、ホストコンピュータ83のCPU101は上記標準処理値CP₀、DI₀、WS₀、BS₀、IX₀、XF₀、LT₀を各処理装置の起動時におけるオペレーションデータCP_x、DI_x、WS_x、BS_x、IX_x、XF_x、LT_xとしてデータメモリ103に保持する(ステップS4)。

次に、ホストコンピュータ83のCPU101はDMAコントローラ106に上記オペレーションデータCP_x、DI_x、WS_x、BS_x、IX_x、XF_x、LT_xを各処理装置に転送を指示する(ステップS5)。

そして、ホストコンピュータ83のCPU101はオペレーションデータCP_x～LT_xの更新サブルーチンを実行するが、これら更新サブルーチンの実行中にタイマーインタラプトが発生すると、更新サブルーチンルーチンの実行を停止して後述するタイマーインタラプトサブルーチンを実行して各処理装置の処理状況を示す上述のレポートデータを受け付ける。したがって、更新されるオペレーションデータは常に最新のレポートデータに

滞留数Ac1が過多か否かを判断する(ステップS54)。

ステップS54の判断結果がYESならば、缶胴の滞留数Ac1が過多なので、その処理数を減少させるためにこのサブルーチンを実行する前のオペレーションデータCP_xから所定の数X₁だけ減算し、その値を再びオペレーションデータCP_xにセットする(ステップS55)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS54の判断結果がNOならば、缶胴の滞留数Ac1が過小なので、その処理数を増加させるためにこのサブルーチンを実行する前のオペレーションデータCP_xに所定の数X₂だけ加算し、その値を再びオペレーションデータCP_xにセットする(ステップS56)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。このようにカップリングプレス14の目標処理数は、アキュムレータの滞留数を参照しつつその下流側処理装置(ドロ

基づき作成されることになる。

ホストコンピュータ83のCPU101は、まず、オペレーションデータCP_xの更新を実行する(ステップS6)。すなわち、第7図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103から水平搬送装置15のアキュムレータ中の缶胴の滞留数Ac1とドロアードアイオニングプレス16の毎分の処理数CPM₀とを読み出す(ステップS51)。

次に、ホストコンピュータ83のCPU101は、缶胴の滞留数Ac1が適正範囲にあるか否かを判断する(ステップS52)。その判断結果がYESならば、缶胴の滞留数Ac1が適正範囲にあるので、オペレーションデータCP_xに上記CPM₀をセットする(ステップS53)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS52の判断結果がNOならば、缶胴の滞留数Ac1が適正範囲外なので、缶胴の

アードアイオニングプレス16)の処理数に応じて増減される。

そして、メインフローでは、ホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータCP_xの転送を指示する(ステップS7)。

次に、ホストコンピュータ83のCPU101はオペレーションデータDI_xの更新サブルーチンを実行する(ステップS8)。すなわち、第8図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103から水平搬送装置23のアキュムレータ中の缶胴の滞留数Ac2と水平搬送装置23の下流側に位置するベースコート24の毎分の処理数CPM₀₀とを読み出す(ステップS61)。

次に、ホストコンピュータ83のCPU101は、缶胴の滞留数Ac2が適正範囲にあるか否かを判断する(ステップS62)。その判断結果がYESならば、缶胴の滞留数Ac2が適正範囲にあるので、オペレーションデータDI_xに上記CP

M_{10} をセットする(ステップS63)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS62の判断結果がNOならば、毎瞬の滞留数Ac2が適正範囲内にないので、ホストコンピュータ83のCPU101は毎瞬の滞留数Ac2が過多か否かを判断する(ステップS64)。

ステップS64の判断結果がYESならば、毎瞬の滞留数Ac2が過多なので、フローアンドアイオニングプレス16の処理数を減少させるためにこのサブルーチンを実行する前のオペレーションデータD1xから所定の数 X_{21} だけ減算し、その値を再びオペレーションデータD1xにセットする(ステップS65)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS64の判断結果がNOならば、毎瞬の滞留数Ac2が過小なので、その処理数を増加させるためにこのサブルーチンを実行する前

のオペレーションデータD1xから所定の数 X_{22} を加算し、その値を再びオペレーションデータD1xにセットする(ステップS66)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。このように、フローアンドアイオニングプレス16の処理数はアキュムレータAc2の滞留数を参照しつつ、ベースコート24の処理数に応じて増減される。

そして、メインフローでは、ホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータD1xの転送を指示する(ステップS9)。

次に、ホストコンピュータ83のCPU101はオペレーションデータWSxの更新サブルーチンを実行する(ステップS10)。

すなわち、第9図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103からベースコート24の毎分の処理数 CPM_{10} とウォッシュ18の現在の処理数 CPM_{10} とウォッシュ18の10分前の

処理数 CPM_{10} から1分前の処理数 CPM_{11} までとを読み出す(ステップS71)。

次に、ホストコンピュータ83のCPU101は、ウォッシュ18の1分前の処理数 CPM_{11} から10分前の処理数 CPM_{10} までの値を合計し、その値をS1にセットする(ステップS72)。

そして、ウォッシュ18の1分前の処理数 CPM_{11} から9分前の処理数 CPM_{10} までの値を新たに、2分前の処理数 CPM_{12} から10分前の処理数 CPM_{10} にセットするとともに、1分前の処理数 CPM_{11} に現在の処理数 CPM_{10} の値をセットする(ステップS73)。

次に、ホストコンピュータ83のCPU101は、ウォッシュ18の新たに設定された1分前の処理数 CPM_{11} から10分前の処理数 CPM_{10} までの値を再び合計しその値をS2にセットする(ステップS74)。

そして、ホストコンピュータ83のCPU101は、S2からS1を減算し、その値を ΔS にセットする(ステップS75)。すなわち、ホスト

コンピュータ83のCPU101はウォッシュ18の1分前から10分間の合計処理数S2と2分前から10分間の合計処理数S1との変化度合を判断する。

次に、ホストコンピュータ83のCPU101は、 ΔS の値が変わらないか($\Delta S = 0$)、それとも増加しているか($\Delta S > 0$)、減少しているか($\Delta S < 0$)を判断する(ステップS76)。その判断結果が $\Delta S = 0$ ならば、ウォッシュ18の1分前から10分間の合計処理数S2と2分前から10分間の合計処理数S1とは変化していないので、ベースコート24の毎分の処理数 CPM_{10} をオペレーションデータWSxにセットする(ステップS77)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS76の判断結果が $\Delta S < 0$ ならば、1分前から10分間の合計処理数S2が2分前から10分間の合計処理数S1より減少している。したがって、ホストコンピュータ83のC

PU101は、その処理数を増加させるために、このサブルーチンを実行する前のオペレーションデータWS_xに所定の数X₃₁だけ加算し、その値を再びオペレーションデータWS_xにセットする（ステップS78）。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

更に、ステップS76の判断結果が $\Delta S > 0$ ならば、1分前から10分間の合計処理数S2が2分前から10分間の合計処理数S1より増加している。したがって、ホストコンピュータ83のCPU101は、その処理数を減少させるために、このサブルーチンを実行する前のオペレーションデータWS_xから所定の数X₃₂だけ減算し、その値を再びオペレーションデータWS_xにセットする（ステップS79）。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

このようにウォッチ18の目標処理数は、アキュムレータの滞留数を参照しつつその下流側処

理装置（ベースコート24）の処理数に応じて増減される。

そして、メインフローではホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータWS_xの転送を指示する（ステップS11）。

そして、ホストコンピュータ83のCPU101はオペレーションデータBS_xの更新サブルーチンを実行する（ステップS12）。すなわち、第10図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103から水平搬送装置27のアキュムレータ中の左側の滞留数Ac3とプリンタ28の毎分の処理数CPM_{pr}とを読み出す（ステップS81）。

次に、ホストコンピュータ83のCPU101は、左側の滞留数Ac3が適正範囲にあるか否かを判断する（ステップS82）。その判断結果がYESならば、左側の滞留数Ac3が適正範囲にあるので、再びオペレーションデータBS_xに上記

CPM_{pr}をセットする（ステップS83）。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS82の判断結果がNOならば、左側の滞留数Ac3が適正範囲外なので、ホストコンピュータ83のCPU101は左側の滞留数Ac3が過多か否かを判断する（ステップS84）。その判断結果がYESならば、左側の滞留数Ac3が過多なので、その処理数を減少させるためにこのサブルーチンを実行する前のオペレーションデータBS_xから所定の数X₄₁だけ減算し、その値を再びオペレーションデータBS_xにセットする（ステップS85）。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS84の判断結果がNOならば、左側の滞留数Ac3が過小なので、その処理数を増加させるためにこのサブルーチンを実行する前のオペレーションデータBS_xに所定の数X₄₂だけ加算し、その値を再びオペレーションデータBS

_xにセットする（ステップS86）。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

このようにベースコート24の目標処理数は、アキュムレータの滞留数を参照しつつその下流側処理装置（プリンタ28）の処理数に応じて増減される。

次に、メインフローではホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータBS_xの転送を指示する（ステップS13）。

そして、ホストコンピュータ83のCPU101はプリンタのエマージェンシーの監視サブルーチンを実行する（ステップS14）。すなわち、第11図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103から水平搬送装置27のアキュムレータ中の左側の滞留数Ac3と水平搬送装置33のアキュムレータ中の右側の滞留数Ac4とを読み出す（ステップS91）。

次に、ホストコンピュータ83のCPU101は、缶胴の滞留数Ac3が適正範囲にあるか否かを判断する(ステップS92)。その判断結果がYESならば、缶胴の滞留数Ac3が適正範囲にあるので、次に缶胴の滞留数Ac4が適正範囲にあるか否かを判断する(ステップS93)。その判断結果がYESならば、缶胴の滞留数Ac4が適正範囲にあるので、すなわち、プリンタ28が正常に動作し、一定の数の缶胴を処理しているので、プリンタ28が異常状態であるか否かを示すフラグEMに「0」をセットする(ステップS94)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS92、またはステップS93の判断結果がNOならば、プリンタ28が異常に動作し、一定の数の缶胴を処理していないので、フラグEMに「1」をセットする(ステップS95)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。このようにプリンタ28のエマージェンシーの

監視は、水平搬送装置27のアキユームレータ中の缶胴の滞留数Ac3と水平搬送装置33のアキユームレータ中の缶胴の滞留数Ac4とに応じて判断される。

次に、メインフローではホストコンピュータ83のCPU101はフラグEMに「1」がセットされているか否かを判断する(ステップS15)。その判断結果がYESならば、プリンタ28が異常に動作し、一定の数の缶胴を処理していないので、全てのオペレーションデータCP_x、DI_x、WS_x、BS_x、IN_x、NF_x、LT_xを減少させ(ステップS16)、ホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータCP_x～LT_xの転送を指示する(ステップS17)。

一方、ステップS15の判断結果がNOならば、プリンタ28が正常に動作し、一定の数の缶胴を処理しているので、ステップS16とステップS17とを実行することなくステップS18を実行する。

次に、ホストコンピュータ83のCPU101はオペレーションデータIN_xの更新サブルーチンを実行する(ステップS18)。すなわち、第12図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103から水平搬送装置33のアキユームレータ中の缶胴の滞留数Ac4とプリンタ28の毎分の処理数CPM₀とを読み出す(ステップS101)。

次に、ホストコンピュータ83のCPU101は、缶胴の滞留数Ac4が適正範囲にあるか否かを判断する(ステップS102)。その判断結果がYESならば、缶胴の滞留数Ac4が適正範囲にあるので、オペレーションデータIN_xに上記CPM₀をセットする(ステップS103)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS102の判断結果がNOならば、缶胴の滞留数Ac4が適正範囲外なので、ホストコンピュータ83のCPU101は缶胴の滞

留数Ac4が過多か否かを判断する(ステップS104)。その判断結果がYESならば、缶胴の滞留数Ac4が過多なので、その処理数を減少させるためにこのサブルーチンを実行する前のオペレーションデータIN_xから所定の数X_{s1}だけ減算し、その値を再びオペレーションデータIN_xにセットする(ステップS105)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS104の判断結果がNOならば、缶胴の滞留数Ac4が過小なので、その処理数を増加させるためにこのサブルーチンを実行する前のオペレーションデータIN_xから所定の数X_{s2}だけ加算し、その値を再びオペレーションデータIN_xにセットする(ステップS106)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

このようにインサイドスプレイ34の目標処理数は、アキユームレータの滞留数を参照しつつその上流側処理装置(プリンタ28)の処理数に応じ

て増減される。

次に、メインフローではホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータNF_xの転送を指示する(ステップS19)。

そして、ホストコンピュータ83のCPU101はオペレーションデータNF_xの更新サブルーチンを実行する(ステップS20)。

すなわち、第13図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103からインサイドスプレイオープン36の今の処理数CPM_{io,x}とインサイドスプレイオープン36の10分前の処理数CPM_{io,e}から1分前の処理数CPM_{io,i}までを読み出す(ステップS121)。

次に、ホストコンピュータ83のCPU101は、インサイドスプレイオープン36の1分前の処理数CPM_{io,i}から10分前の処理数CPM_{io,e}までの値を合計し、その値をS3にセットする(ステップS122)。

とも増加しているか($\Delta S > 0$)、減少しているか($\Delta S < 0$)を判断する(ステップS126)。その判断結果、 $\Delta S = 0$ ならば、インサイドスプレイオープン36の1分前から10分間の合計処理数S4と2分前から10分間の合計処理数S3とは変化していないので、10分前の処理数CPM_{io,e}をオペレーションデータNF_xにセットする(ステップS127)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS126の判断結果、 $\Delta S < 0$ ならば、1分前から10分間の合計処理数S4が2分前から10分間の合計処理数S3より減少している。したがって、ホストコンピュータ83のCPU101は、その処理数を増加させるために、このサブルーチンを実行する前のオペレーションデータNF_xに所定の数X₆₂だけ加算し、その値を再びオペレーションデータNF_xにセットする(ステップS128)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メイン

そして、インサイドスプレイオープン36の1分前の処理数CPM_{io,i}から9分前の処理数CPM_{io,e}までの値を新たに、2分前の処理数CPM_{io,e}から10分前の処理数CPM_{io,i}にセットするとともに、1分前の処理数CPM_{io,i}に今の処理数CPM_{io,x}の値をセットする(ステップS123)。

次に、ホストコンピュータ83のCPU101は、インサイドスプレイオープン36の1分前の処理数CPM_{io,i}から10分前の処理数CPM_{io,e}までの値を再び合計しその値をS4にセットする(ステップS124)。

そして、ホストコンピュータ83のCPU101は、S4からS3を減算し、その値を ΔS にセットする(ステップS125)。すなわち、ホストコンピュータ83のCPU101はインサイドスプレイオープン36の1分前から10分間の合計処理数S4と2分前から10分間の合計処理数S3との変化の具合を確認している。

次に、ホストコンピュータ83のCPU101は、 ΔS の値が変わらないか($\Delta S = 0$)、それ

フローに戻る。

更に、ステップS126の判断結果、 $\Delta S > 0$ ならば、1分前から10分間の合計処理数S4が2分前から10分間の合計処理数S3より増加している。したがって、ホストコンピュータ83のCPU101は、その処理数を減少させるために、このサブルーチンを実行する前のオペレーションデータNF_xから所定の数X₆₁だけ減算し、その値を再びオペレーションデータNF_xにセットする(ステップS129)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

このようにネッカフランジ38の目標処理数は、アキュムレータの滞留数を参照しつつその上流側処理装置(インサイドスプレイオープン36)の処理数に応じて増減される。

そして、メインフローではホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータNF_xの転送を指示する(ステップS21)。

次に、ホストコンピュータ83のCPU101はオペレーションデータLT_xの更新サブルーチンを実行する(ステップS22)。すなわち、第14図のフローチャートに示すように、まず、ホストコンピュータ83のCPU101は、データメモリ103から水平搬送装置40のアキუმレータ中の缶胴の滞留数A_{c5}とネッカーフランジヤ38の毎分の処理数CPM_{nr}を読み出す(ステップS111)。

次に、ホストコンピュータ83のCPU101は、缶胴の滞留数A_{c5}が適正範囲にあるか否かを判断する(ステップS112)。その判断結果がYESならば、缶胴の滞留数A_{c5}が適正範囲にあるので、オペレーションデータLT_xに上記CPM_{nr}をセットする(ステップS113)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS112の判断結果がNOならば、缶胴の滞留数A_{c5}が適正範囲外なので、ホストコンピュータ83のCPU101は缶胴の滞

りて増減される。

次に、メインフローではホストコンピュータ83のCPU101はDMAコントローラ106に上記更新されたオペレーションデータLT_xの転送を指示する(ステップS23)。

そして、ホストコンピュータ83のCPU101は、その他のオペレーションデータを作成し、所定の処理装置に転送する(ステップS24)。

次に、ホストコンピュータ83のCPU101は、上記オペレーションデータCP_x、DI_x、WS_x、BS_x、IN_x、NF_x、LT_xと上記処理数CPM_{cs}、CPM_{cs}、CPM_{cs}、CPM_{cs}、CPM_{cs}、CPM_{cs}、CPM_{cs}、CPM_{cs}とをプリンタ108に出力する要求があったか否かを判断する(ステップS25)。その判断結果がYESならば、ホストコンピュータ83のCPU101はDMAコントローラ106に上記オペレーションデータCP_x、LT_xと上記処理数CPM_{cs}～CPM_{cs}とをプリンタ108に転送する指示をする(ステップS26)。

滞留数A_{c5}が過多か否かを判断する(ステップS114)。その判断結果がYESならば、缶胴の滞留数A_{c5}が過多なので、その処理数を減少させるためにこのサブルーチンを実行する前のオペレーションデータLT_xから所定の数X_{r1}だけ減算し、その値を再びオペレーションデータLT_xにセットする(ステップS115)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

一方、ステップS114の判断結果がNOならば、缶胴の滞留数A_{c5}が過小なので、その処理数を増加させるためにこのサブルーチンを実行する前のオペレーションデータLT_xから所定の数X_{r2}だけ加算し、その値を再びオペレーションデータLT_xにセットする(ステップS116)。そして、ホストコンピュータ83はこのサブルーチンの実行を終了し、メインフローに戻る。

このようにライトテスト41の目標処理数は、アキუმレータの滞留数を参照しつつその上流側処理装置(ネッカーフランジヤ38)の処理数に

一方、ステップS25の判断結果がNOならば、ステップS26を実行することなく、ホストコンピュータ83のCPU101はDMAコントローラ106に上記オペレーションデータCP_x～LT_xと上記処理数CPM_{cs}～CPM_{cs}とをビデオメモリ103に転送する指示をする(ステップS27)。そして、ホストコンピュータ83のCPU101は、上記オペレーションデータCP_x～LT_xと上記処理数CPM_{cs}～CPM_{cs}とをCRT109に表示する指示をする(ステップS27)。

そして、ホストコンピュータ83のCPU101は、ステップS6～ステップS27のフローを繰り返して実行する。

また、ホストコンピュータ83のCPU101は、公知のソフトウェアでタイマーを実現しており、一定時間経過する毎にインタラプトが発生する。なお、上記インタラプトが発生すると、ホストコンピュータ83のCPU101はDMAコントローラ106にID_Xフラグで指定されたスレーブコンピュータ77～79から上記オペレーシ

ョンデータCPU₈₃～LT₈₃の転送を命じ、その終了の報告後に再スタートする。

したがって、ホストコンピュータ83のCPU101は、メインフローを実行しているがタイマ—インタラプトが発生すると、メインフローの実行を中断して以下のプログラムを実行することになる。

まず、第6図に示すように、ホストコンピュータ83のCPU101は、スレーブコンピュータを指定するIDXフラグに「0」がセットされているか否かを判断する(ステップS31)。その判断結果がYESならば、スレーブコンピュータ77が指定されているので次のスレーブコンピュータ78を指定するためにIDXフラグに「1」をセットする(ステップS32)。

次に、ホストコンピュータ83のCPU101は、スレーブコンピュータ77にレポートデータを要求する(ステップS33)。

そして、ホストコンピュータ83のCPU101は、上記指定されたスレーブコンピュータ77

から送付されたレポートデータをインターフェース107からデータメモリへ転送するようにDMAコントローラ106に指示する(ステップS34)。

一方、ステップS31の判断結果がNOならば、スレーブコンピュータ77が指定されていないのでステップS35に進む。

次に、ステップS35では、ホストコンピュータ83のCPU101は、スレーブコンピュータを指定するIDXフラグに「1」がセットされているか否かを判断する(ステップS35)。その判断結果がYESならば、スレーブコンピュータ78が指定されているので次のスレーブコンピュータ79を指定するためにIDXフラグに「2」をセットする(ステップS36)。

次に、ホストコンピュータ83のCPU101は、スレーブコンピュータ78にレポートデータを要求する(ステップS37)。

そして、ホストコンピュータ83のCPU101は、上記レポートデータをインターフェース1

07からデータメモリへ転送するようにDMAコントローラ106に指示する(ステップS38)。

一方、ステップS35の判断結果がNOならば、スレーブコンピュータ78が指定されていないのでステップS39を進む。

次に、ステップS39では、ホストコンピュータ83のCPU101は、スレーブコンピュータを指定するIDXフラグに「2」がセットされているか否かを判断する(ステップS39)。その判断結果がYESならば、スレーブコンピュータ79が指定されているので再びスレーブコンピュータ77を指定するためにIDXフラグに「0」をセットする(ステップS40)。

次に、ホストコンピュータ83のCPU101は、スレーブコンピュータ79にレポートデータを要求する(ステップS41)。

そして、ホストコンピュータ83のCPU101は、上記レポートデータをインターフェース107からデータメモリへ転送するようにDMAコントローラ106に指示する(ステップS42)。

一方、ステップS39の判断結果がNOならば、スレーブコンピュータ79が指定されていないのでその他のジョブを実行し、メインフローに戻る。

このように、ホストコンピュータ83はスレーブコンピュータ79を介して各装置の処理状況を監視しているため、オペレータの数を減らすことができ、その結果、生産コストを下げるができる。

また、ホストコンピュータ83はスレーブコンピュータ77、およびスレーブコンピュータ78を介して各装置の処理速度の設定変更作業を行っているため、処理速度の設定変更作業の時間を短縮させることができ、その結果、製缶ラインの稼働率を高めることができる。

さらに、ホストコンピュータ83がスレーブコンピュータ79を介して各装置の処理状況を監視、ならびに、スレーブコンピュータ77とスレーブコンピュータ78とを介して処理速度の設定変更作業を行っているため、上述したように缶のクラッシュ等のトラブルによって一部の装置の処理速

度が低下しても素早く全装置の再設定を行うことができるので、その結果、トラブルに対応する的確な処理が容易である。

4. 図面の簡単な説明

第1図は本発明の一実施例に係る製缶ラインの制御システムを機能実現手段によって示すブロック図、第2A図～第2C図は本発明の一実施例に係る製缶ラインの制御システムを概略して示す側面図、第3図は水平搬送装置の平面図、第4図は製缶ラインの制御システムのホストコンピュータのハードウェアの構成を示すブロック図、第5A図～第5B図はホストコンピュータのメインフローチャート、第6図はホストコンピュータのタイマインタラプトのフローチャート、第7図～第15図はホストコンピュータの各サブルーチンのフローチャートである。

1 基準装置、

2 第1装置、

3 第2装置、

4 第1アキュムレータ、

5 第1制御手段、

6 第3装置、

7 第4装置、

8 第2アキュムレータ、

9 第2制御手段、

14 カッピングプレス、

16 ドローアンドアイオニングプレス、

18 ウォッシャ、

24 ベースコータ、

28 プリンタ、

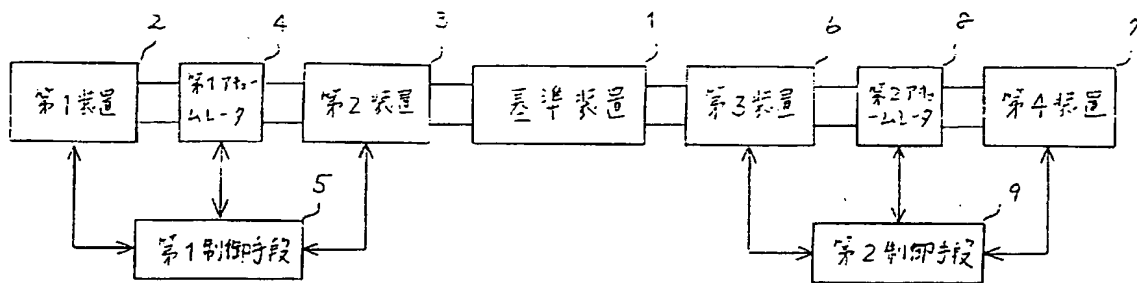
34 インサイドスプレイ、

36 インサイドスプレイオープン、

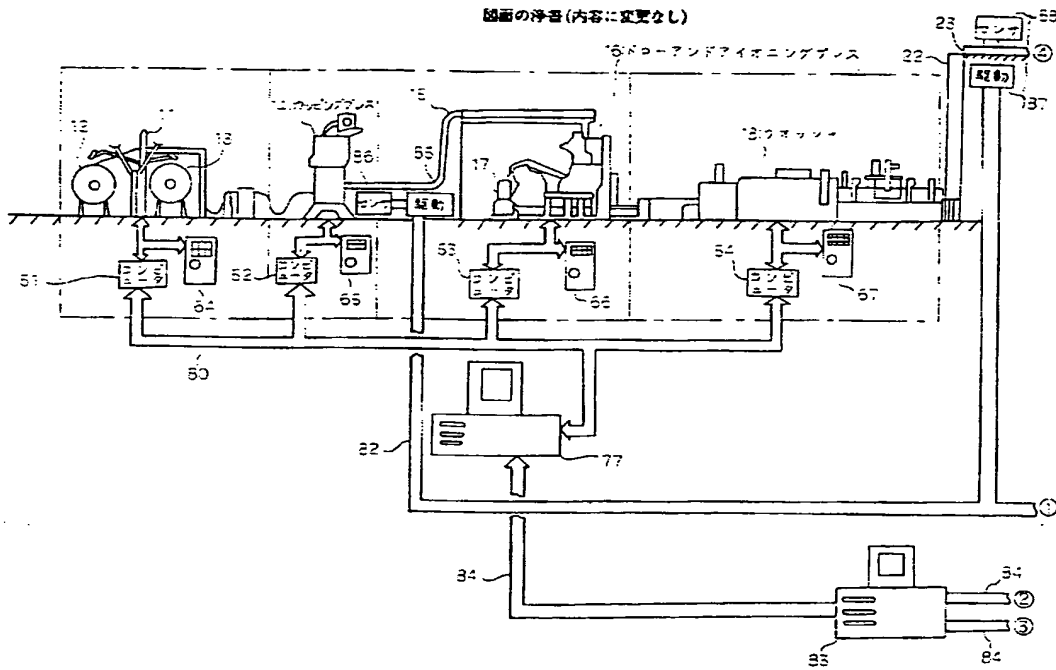
38 ネッカーフランジヤ、

77～79 スレーブコンピュータ、

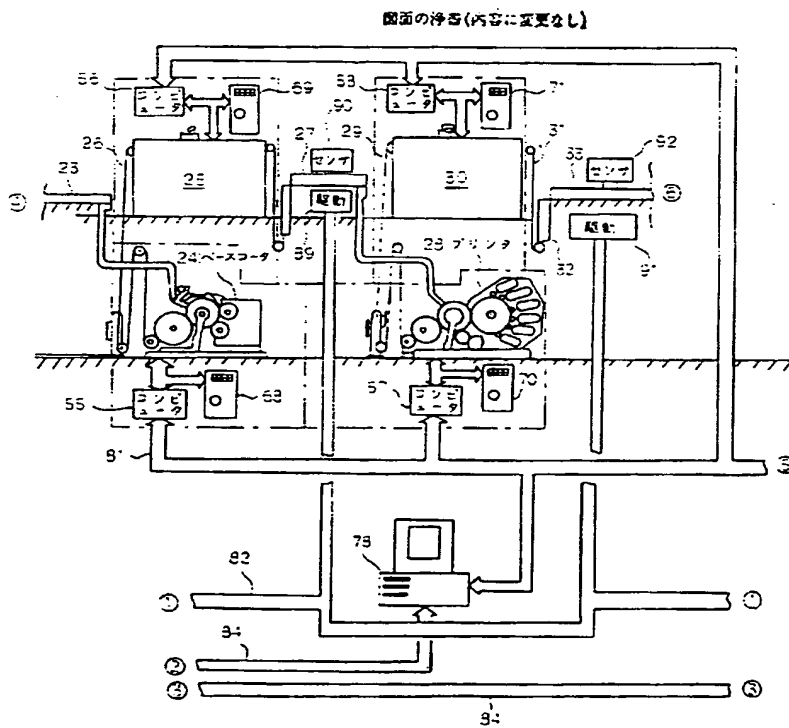
83 ホストコンピュータ。



第1図

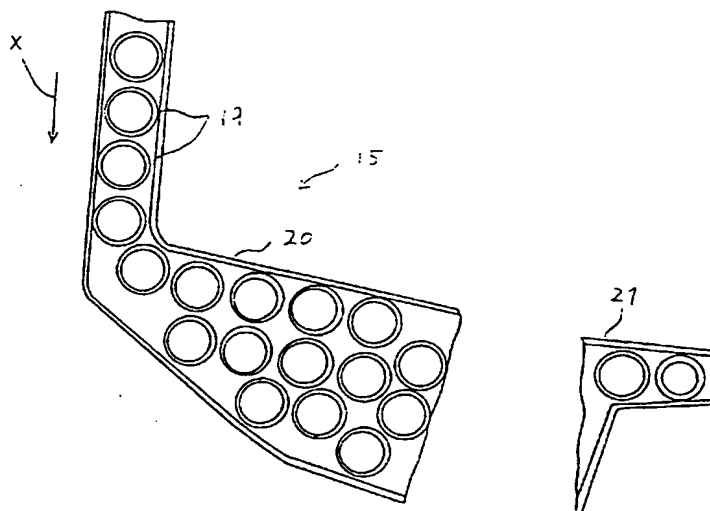
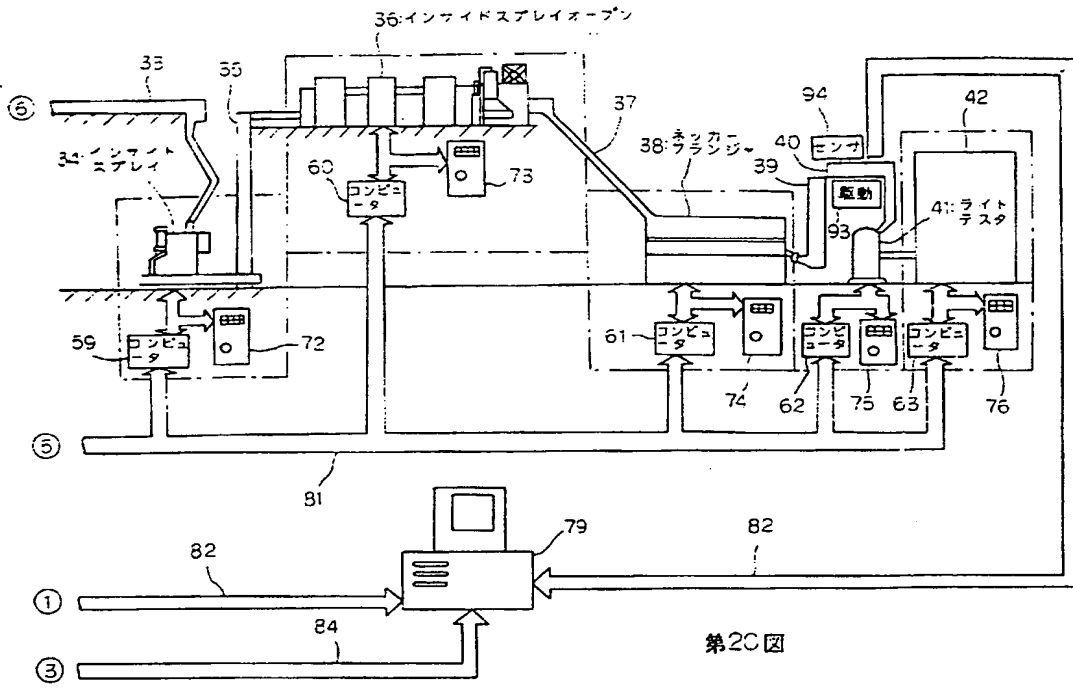


第24図



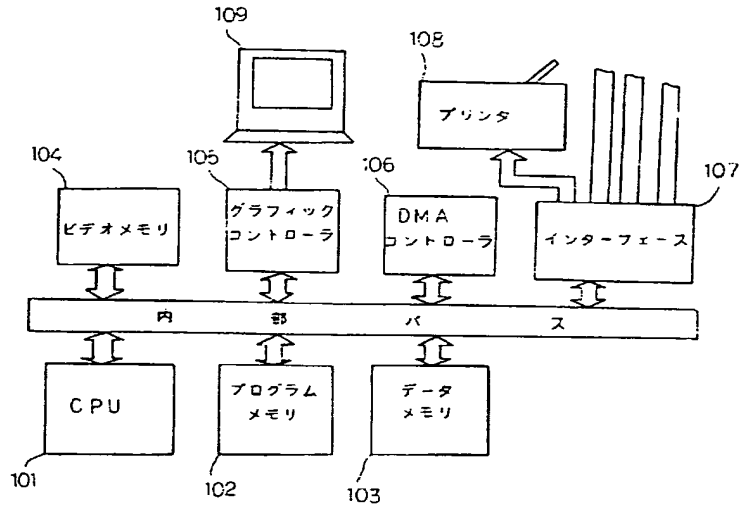
第25図

図面の浄化(内容に変更なし)

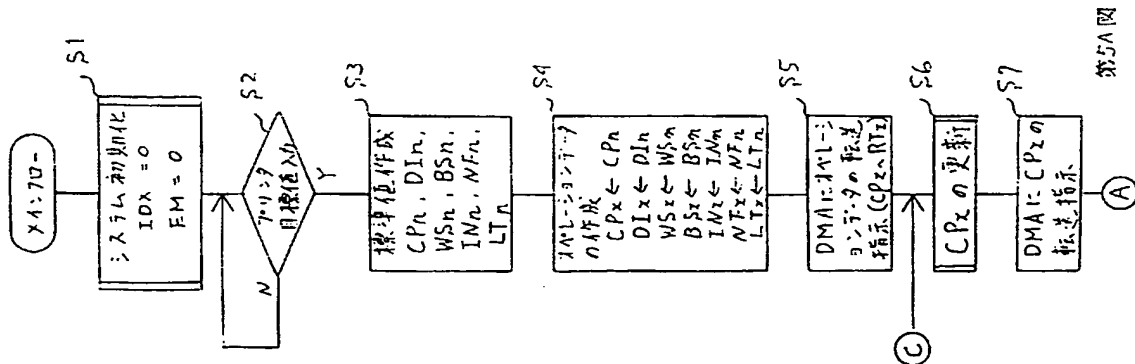
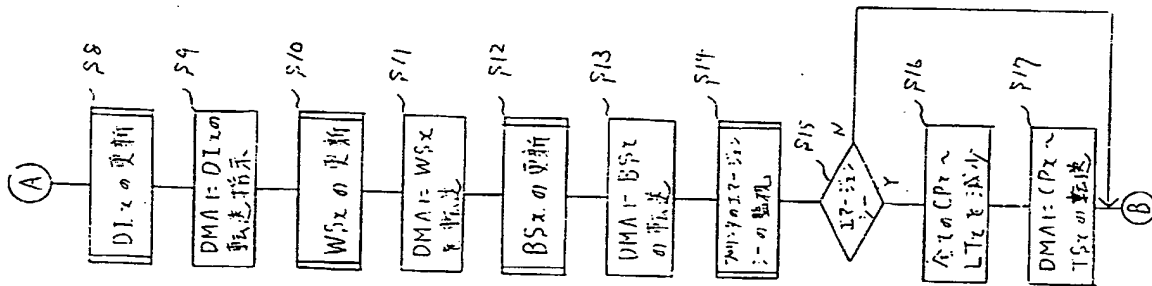


第3図

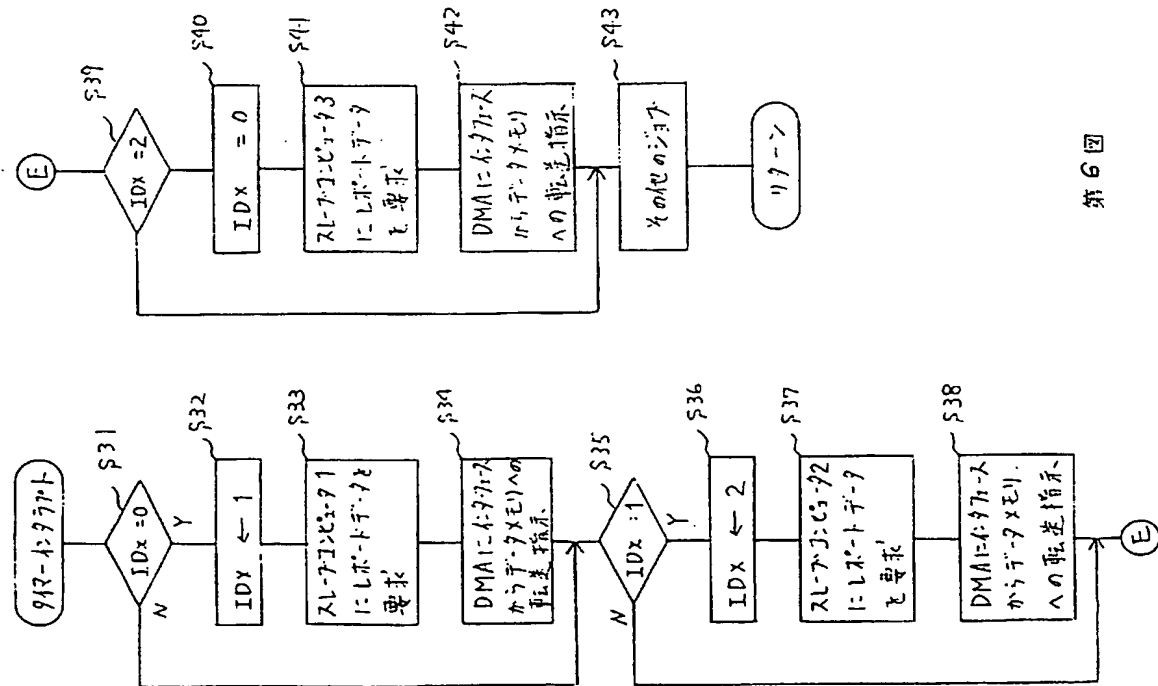
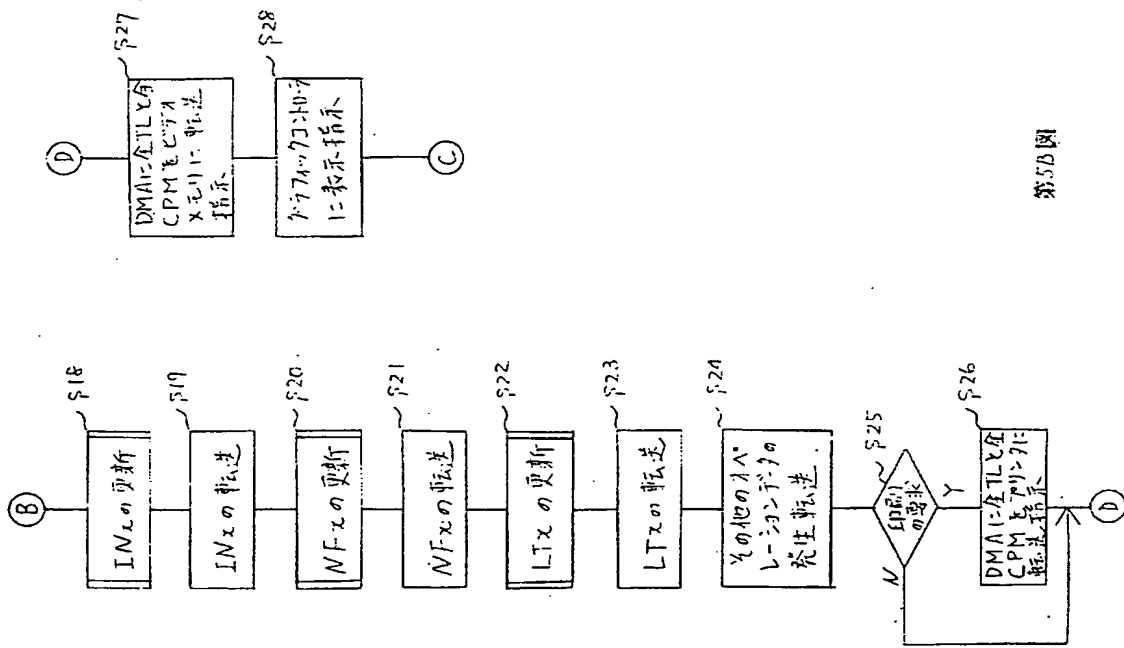
図面の符号(内容に変更なし)

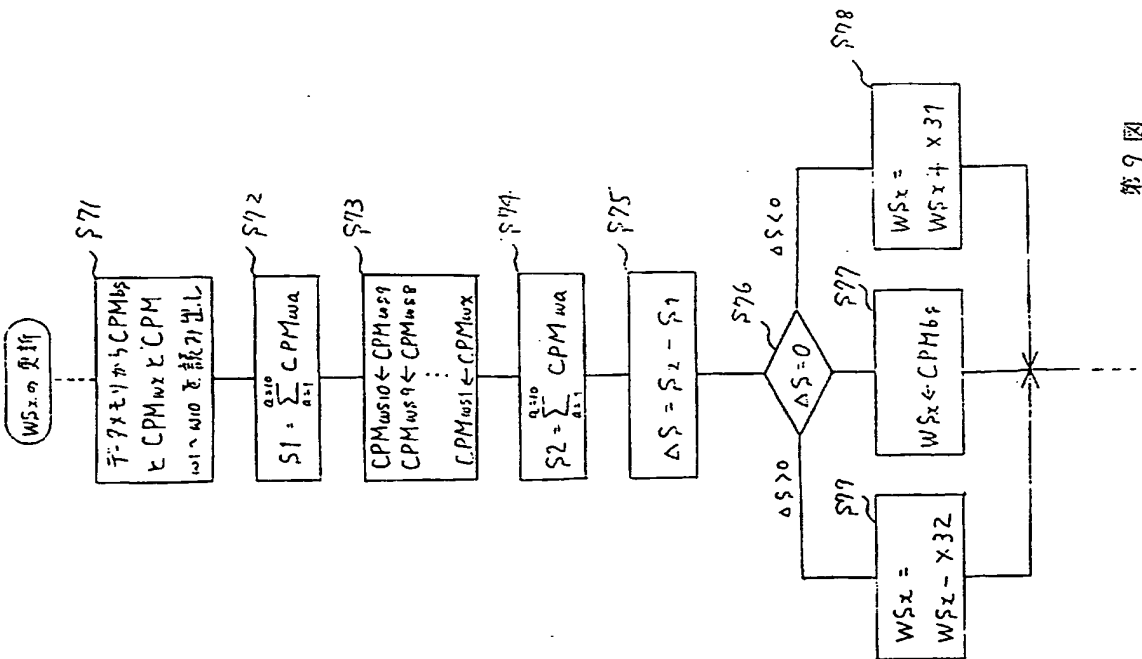


第4図

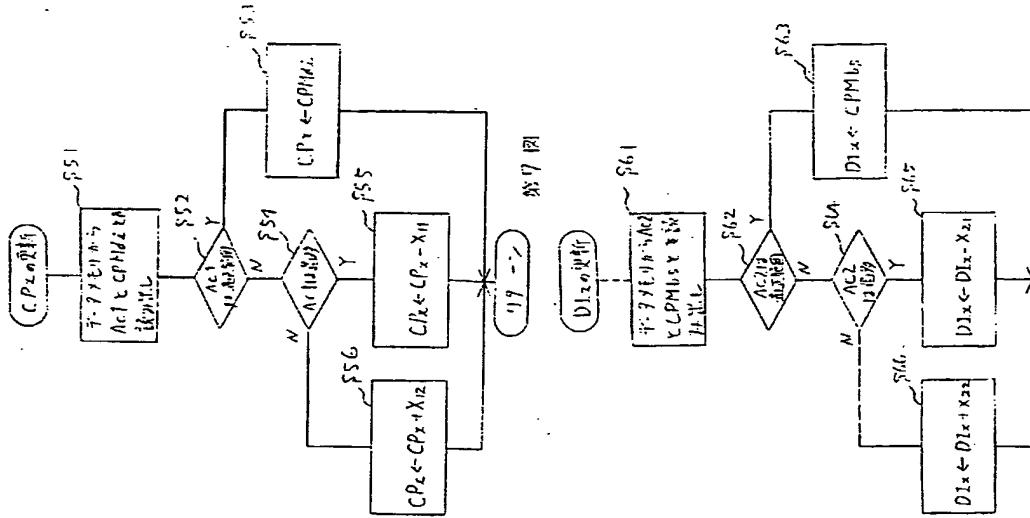


第5A図



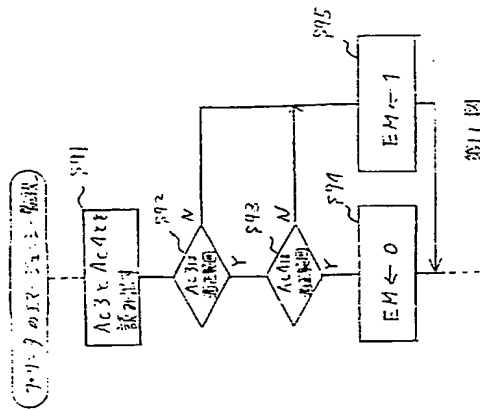
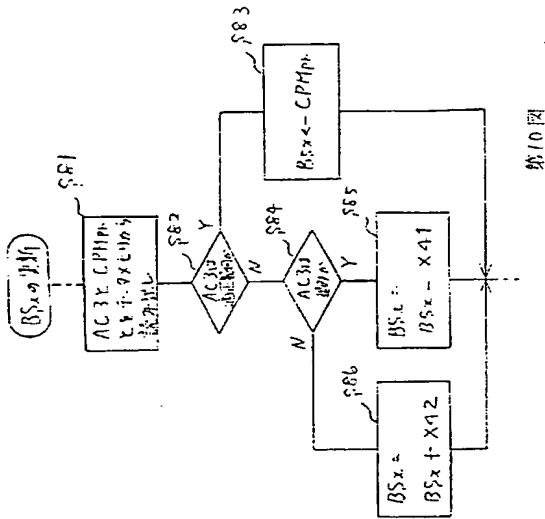
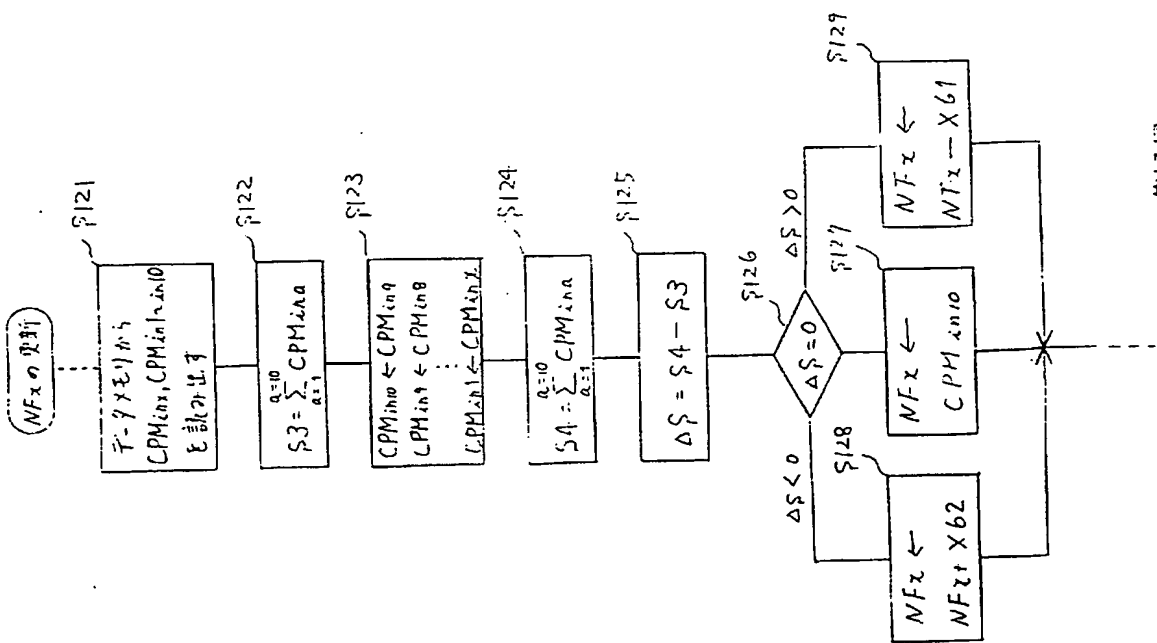


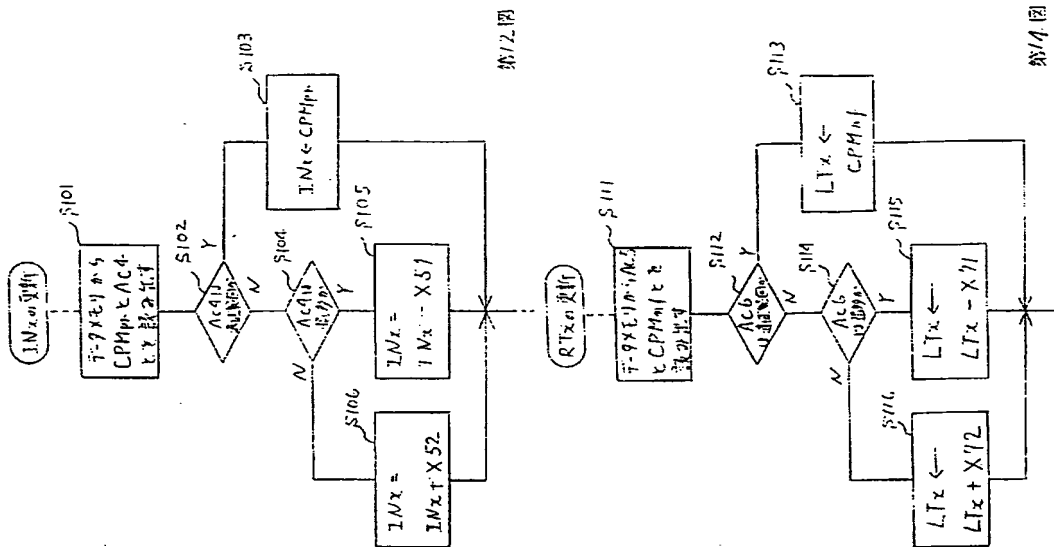
第 9 図



第 7 図

第 7 図





手続補正書 (方式)

平成3年4月9日

特許庁長官 殿

1. 事件の表示
平成2年特許願第323758号
2. 発明の名称
製造ラインの処理速度制御システム
3. 補正をする者
事件との関係 特許出願人

住所 東京都千代田区大手町一丁目6番1号
名称 (625)三菱マテリアル株式会社
代表者 藤村 正哉

4. 代理人
住所 〒150
東京都渋谷区神山町40番4号
大月マンション1階
氏名 (8917) 森井 清一 (外1名)

5. 補正命令の日付

平成3年3月12日

6. 補正の対象
図面の第2A図、第2B、第2C図および第4図。
7. 補正の内容
願書に最初に添付した第2A図、第2B図、第2C図および第4図の浄書 (内容に変更なし)。

特開平4-365709 (23)

手続補正書 (方式)

平成4年7月6日

特許庁長官 殿

1. 事件の表示
平成2年特許願第323758号
2. 発明の名称
製缶ラインの処理速度制御システム

3. 補正をする者
事件との関係 特許出願人
住 所 東京都千代田区大手町一丁目6番1号
名 称 (626)三菱マテリアル株式会社
代表者 藤村 正哉

4. 代理人
居 所 〒150
東京都渋谷区神山町40番4号
大月マンション1階
電話 03-3469-7418
氏 名 (8917)弁理士 桑井 清一
(外1名)

5. 補正命令の日付 平成4年5月18日
(発送日 平成4年6月30日)

6. 補正の対象
明細書の「図面の簡単な説明」の欄。

7. 補正の内容
明細書第55頁第14行目～第15行目に、
「第7図～第15図」とあるのを、「第7図～
第14図」と補正する。

以 上



